Fusion Discrete Logarithm Problems

Martin Schaffer, NXP Semiconductors
Joint work with Stefan Rass and Patrick Horster

1. Introduction

Many cryptosystems are proven to be secure under a particular computational assumption, such as RSA [8] for instance, resting its security on the difficulty of the Factoring Problem. Many others, such as ElGamal [4], are based on the Discrete Logarithm Problem [6] and several other related problems on which the focus of this abstract lies. Henceforth, we consider a group $\mathbb{G}_q$ of prime order q. Therein, the *Discrete Logarithm Problem* (DLP) is the following: given $y, g \in \mathbb{G}_q$, $g \neq 1$, and q, find $x \in \mathbb{Z}_q$, such that $y = g^x$. The integer x is called the *discrete logarithm* of y to the base g, denoted as $dlog_g(y)$. The problem of constructing $g^{x_1 x_2}$ solely from (g^{x_1}, g^{x_2}) is known as the *Diffie-Hellman Problem* (DHP) [3]. To decide, whether a given triple $(y_1, y_2, y_3) \in \mathbb{G}_q^3$ is of the form $(g^{x_1}, g^{x_2}, g^{x_1 x_2})$ is known as the *Decision Diffie-Hellman Problem* (DDP) [1]. Obviously, solving the DLP gives trivial solutions to the DHP and the DDP, respectively. Similarly, solving the DHP leads to an efficient solution of the DDP. The converse directions are less obvious. Some results concerning the relations between the DLP and the DHP can be found in [5]. However, it is still an open question (of course it depends on the group used), how solutions of the DDP can be used to solve the DHP. This is also known as the *Gap Diffie-Hellman Problem* [7]. Several cryptosystems are based on the DLP, DHP or DDP. The ElGamal encryption scheme, for instance, is semantically secure under the assumption that solving the DHP is hard. Moreover, under the assumption that the DDP is hard, it is guaranteed that upon two given ciphertexts, it is not efficiently possible to decide, if both contain the same plaintext. Unfortunately, the ElGamal encryption scheme is insecure against chosen ciphertext attacks [11]. The Cramer-Shoup cryptosystem [2] overcomes this drawback, while resting its security on the DDP.

We recently developed a particular algebraic structure together with a function ξ that shares fundamental properties with ordinary exponentiation in $\mathbb{G}_q$. The main difference is that ξ requires pairs of elements of $\mathbb{G}_q$ in the basis and pairs of elements of $\mathbb{Z}_q$ in the exponent. ξ is designed such that its execution uniformly includes all four input-elements for the computation of the output, a pair of elements of $\mathbb{G}_q$. We call this inclusion *fusion*. As ξ shares basic properties with exponentiation, we call it *fusion-exponentiation* and define the Fusion DLP (FDLP), the Fusion DHP (FDHP) and the Fusion DDP (FDDP) in the usual way.

The remainder of this abstract is organized as follows. Firstly, we collect the basic properties of ordinary exponentiation, as these make up the minimum requirements for the fusion-exponentiation function ξ, being introduced afterwards through a constructive approach. This enables us to define the FDLP, FDHP and FDDP. Finally, we sketch the problem of showing the computational equivalence of the DDP and FDDP and discuss its advantageous effect on related cryptosystems.

2. Ordinary Exponentiation

An element $g \in \mathbb{G}_q$ may be raised to the power of $x \in \mathbb{Z}_q$ by multiplying g with itself x-times. For all $g, h \in \mathbb{G}_q$ and $x, y \in \mathbb{Z}_q$ we have:

$$(g^x)^y = g^{xy} \tag{1}$$

$$g^{x+y} = g^x g^y \tag{2}$$

$$(gh)^x = g^x h^x \tag{3}$$

Furthermore, $g^0 = 1$ and $g^{-x} = (g^x)^{-1}$. The properties stated above are fundamental for realizing (basic) discrete-logarithm-based cryptosystems.

3. Fusion-Exponentiation

In the fusion-setting, we define exponents as pairs of integers of $\mathbb{Z}_q$. It would be convenient to have the exponents of our generalized exponentiation come from a field (in fact a commutative ring with 1 would suffice, but a field gives rise to a wider class of possible applications), while in the basis we only require a group. A natural choice for the source of the exponents is thus a field of order $p = q^2$, which is easily constructed by choosing $q \equiv 3 \pmod 4$, and $\mathbb{F}_p := \mathbb{Z}_q[X]/(X^2+1)$. For simplicity, we sometimes abbreviate a pair (a, b) by a sans-serif font letter, e.g. x.

In the following, we get started by deriving property (1) for our new exponentiation where exponents consist of two integers. Let us define a simple form, taking a pair in the exponent, as

$$g^{\mathsf{x}} = g^{(c,d)} := (g^c, g^d), \tag{4}$$

where $\mathsf{x} \in \mathbb{F}_p$, $\mathsf{x} = (c, d)$ and $g \in \mathbb{G}_q \setminus \{1\}$. Suppose we are given a term g^{x} according to the convention (4), and we wish to find $(g^{\mathsf{x}})^{\mathsf{y}}$ such that the result equals g^{xy}, i.e. we need to calculate the latter term given only $g^{\mathsf{x}} = (g^c, g^d)$ and $\mathsf{y} = (e, f)$, where $\mathsf{y} \in \mathbb{F}_p$. This is easily done, as

$$\begin{aligned} g^{\mathsf{xy}} &= g^{(c,d)(e,f)} = g^{(ce-df,cf+de)} \\ &\overset{(4)}{=} (g^{ce-df}, g^{cf+de}) \\ &\overset{(1),(2)}{=} ((g^c)^e (g^d)^{-f}, (g^c)^f (g^d)^e). \end{aligned} \tag{5}$$

Hence, we can define the exponentiation

$$(g^{\mathsf{x}})^{\mathsf{y}} = (g^{(c,d)})^{(e,f)} \overset{(4)}{=} (g^c, g^d)^{(e,f)}$$

through (5) as

$$(g^{\mathsf{x}})^{\mathsf{y}} := ((g^c)^e (g^d)^{-f}, (g^c)^f (g^d)^e) = g^{\mathsf{xy}}.$$

Because g is a generator of $\mathbb{G}_q$, we can write any two elements $a, b \in \mathbb{G}_q$ as $a = g^c$, $b = g^d$ for some integers $c, d \in \mathbb{Z}_q$. Substituting the powers of g in (5), we obtain

$$(g^{\mathsf{x}})^{\mathsf{y}} = (a^e b^{-f}, a^f b^e), \tag{6}$$

and the *fusion-exponentiation* $\xi : \mathbb{G}_p \times \mathbb{F}_p \to \mathbb{G}_p$ is found by observing that by (4), any pair $(a, b) \in \mathbb{G}_q \times \mathbb{G}_q =: \mathbb{G}_p$ can be written using powers of g as (g^c, g^d), such that with g^{x} being represented by (a, b), from (6) we arrive at the definition

$$(a, b)^{(e,f)} := (a^e b^{-f}, a^f b^e),$$

satisfying (1) by construction. Since $\mathbb{G}_p$ is simply the outer product of $\mathbb{G}_q$ with itself, it is a group with component-wise multiplication, i.e. $(a_1, b_1)(a_2, b_2) = (a_1 a_2, b_1 b_2)$ for $(a_1, b_1), (a_2, b_2) \in \mathbb{G}_p$. Having this together with $\mathbb{F}_p$ being a field, the properties (2) and (3) can be shown to hold [10]. To keep computing discrete logarithms hard, it is intrinsic that exponentiation is done using a basis of large order. In $\mathbb{G}_q$, every element $g \neq 1$ has maximum order q. An analogous result can be shown for $\mathbb{G}_p$ regarding fusion-exponentiation. Every element $\mathsf{g} \neq (1, 1)$ can be used to generate $\mathbb{G}_p$ using fusion-exponentiation. A proof can be found in [10].

4. Fusion Discrete Logarithm Problems

The FDLP, FDHP and FDDP can be defined analogously to the DLP, DHP and DDP. The *Fusion Discrete Logarithm Problem* is the following: given $\mathsf{y}, \mathsf{g} \in \mathbb{G}_p$, $\mathsf{g} \neq (1, 1)$, and q, find an element $\mathsf{x} \in \mathbb{F}_p$, such that $\mathsf{y} = \mathsf{g}^{\mathsf{x}}$. We call x the *fusion discrete logarithm* of y to the base g denoted as $fdlog_{\mathsf{g}}(\mathsf{y})$. The *Fusion Diffie-Hellman Problem* is the following: given $\mathsf{y}_1, \mathsf{y}_2, \mathsf{g} \in \mathbb{G}_p$, $\mathsf{g} \neq (1, 1)$, and q, find $\mathsf{y}_3 \in \mathbb{G}_p$, such that $(\mathsf{y}_1, \mathsf{y}_2, \mathsf{y}_3)$ is of the form $(\mathsf{g}^{\mathsf{x}_1}, \mathsf{g}^{\mathsf{x}_2}, \mathsf{g}^{\mathsf{x}_1\mathsf{x}_2})$, for some $\mathsf{x}_1, \mathsf{x}_2 \in \mathbb{F}_p$. To decide whether a given triple is of this form, we define as the *Fusion Decision Diffie-Hellman Problem*. In addition to these three problems, one can define several sub-problems. For instance, given $\mathsf{y}, \mathsf{g} \in \mathbb{G}_p$, $\mathsf{g} \neq (1, 1)$, find x_1 or x_2, where $(x_1, x_2) \in \mathbb{F}_p$, such that $\mathsf{y} = \mathsf{g}^{(x_1,x_2)}$. Or another variant: given x_1 (resp. x_2) in addition, find x_2 (resp. x_1). Interestingly, all sub-problems of this form are equally hard to solve and not easier than the above defined ones. This is due to the fact that both parts of the exponent are uniformly included during computations [9, 10].

5. Open Problem

Solving the FDDP leads to a trivial solution of the DDP: let $\mathcal{O}_{\text{FDDP}}$ be an oracle, which on input $\mathsf{y}_1, \mathsf{y}_2, \mathsf{y}_3, \mathsf{g} \in \mathbb{G}_p$, $\mathsf{g} \neq (1,1)$, efficiently decides if $fdlog_{\mathsf{g}}(\mathsf{y}_3) = fdlog_{\mathsf{g}}(\mathsf{y}_1) fdlog_{\mathsf{g}}(\mathsf{y}_2)$. The DDP can then be solved as follows: given $(y_1, y_2, y_3) \in \mathbb{G}_q^3$ and $g \in \mathbb{G}_q$, $g \neq 1$, query

$$\mathcal{O}_{\text{FDDP}}((y_1,1),(y_2,1),(y_3,1),(g,1)),$$

which returns true if and only if (y_1, y_2, y_3) is of the form $(g^{x_1}, g^{x_2}, g^{x_1 x_2})$, for some $x_1, x_2 \in \mathbb{Z}_q$, because

$$fdlog_{(g,1)}((y_i,1)) = (dlog_g(y_i), 0) =: (x_i, 0)$$

for $i = 1, 2, 3$ and

$$(x_1, 0)(x_2, 0) = (x_3, 0)$$

if and only if $x_3 = x_1 x_2$, and false otherwise. For the reverse direction let $\mathcal{O}_{\text{DDP}}$ be an oracle that on input $y_1, y_2, y_3, g \in \mathbb{G}_q$, $g \neq 1$, efficiently decides if $dlog_g(y_3) = dlog_g(y_1) dlog_g(y_2)$. We have

$$\mathsf{g}^{\mathsf{x}_1} = (g_1^{x_{11}} g_2^{-x_{12}}, g_1^{x_{12}} g_2^{x_{11}}) \quad \text{and} \quad \mathsf{g}^{\mathsf{x}_2} = (g_1^{x_{21}} g_2^{-x_{22}}, g_1^{x_{22}} g_2^{x_{21}}),$$

where $\mathsf{x}_1 = (x_{11}, x_{12})$, $\mathsf{x}_2 = (x_{21}, x_{22})$ and $(g_1, g_2) \in \mathbb{G}_p$, and want to decide if $\mathsf{g}^{\mathsf{x}_3}$ is of the form

$$(g_1^{x_{11}x_{21} - x_{12}x_{22}} g_2^{-x_{11}x_{22} - x_{12}x_{21}}, g_1^{x_{11}x_{22} + x_{12}x_{21}} g_2^{x_{11}x_{21} - x_{12}x_{22}}).$$

The problem arises from the oracle's inability to provide more than true/false-decisions. All of our current approaches to give an efficient reduction to the DDP end up in the necessity to have an oracle for solving the DHP. Such an oracle, however, is not available for the reduction.

6. Conclusion

The above stated open problem yields an interesting conjecture: if the computational equivalence between the DDP and FDDP cannot be shown, then the FDDP seems to be a stronger problem than the DDP. Thus, if the DDP is efficiently solved directly (i.e. without solving the DLP or DHP), then related cryptosystems like ElGamal or Cramer-Shoup will become vulnerable. However, if our conjecture remains unrefuted, then such cryptosystems will still remain secure within the fusion-setting. Certainly, if the computational equivalence between the DDP and FDDP is shown, then fusion-exponentiation will not bring an advantage concerning security. Notice also that the bit-security is always associated to the same prime q, no matter which setting is used. But it offers some interesting algebraic properties and perhaps some security features in a different manner.

References

[1] D. Boneh. The Decision Diffie-Hellman Problem. In *Proceedings of the Third International Symposium on Algorithmic Number Theory – ANTS-III*, LNCS vol. 1423, p. 48–63. Springer, 1998.

[2] R. Cramer and V. Shoup. A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack. In *Advances in Cryptology (CRYPTO'98)*, LNCS vol. 1462, p. 13–25. Springer, 1998.

[3] W. Diffie and M. E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.

[4] T. ElGamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In *Advances in Cryptology (CRYPTO'84)*, LNCS vol. 196, p. 10–18. Springer, 1985.

[5] U. Maurer and S. Wolf. The Relationship Between Breaking the Diffie-Hellman Protocol and Computing Discrete Logarithms. *SIAM Journal on Computing*, 28(5):1689–1721, 1999.

[6] K. S. McCurley. The Discrete Logarithm Problem. In *Cryptology and Computational Number Theory*, vol. 42, p. 49–74. American Mathematical Society, 1990.

[7] T. Okamoto and D. Pointcheval. The Gap-Problems: A New Class of Problems for the Security of Cryptographic Schemes. In *Proceedings of PKC'01*, LNCS vol. 1992, p. 104–118. Springer, 2001.

[8] R. L. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[9] M. Schaffer. *Key Aspects of Random Number Generation*. VDM-Verlag, 2008.

[10] M. Schaffer and S. Rass. Secure Collision-Free Distributed Key Generation for Discrete-Logarithm-Based Threshold Cryptosystems. In *Contributions to General Algebra 18*, p.159–174, Heyn-Verlag, 2008.

[11] V. Shoup. Why Chosen Ciphertext Security Matters. Research Report RZ 3076 (#93122), IBM Research Zurich, 1998.